

CLAIMS:

We claim:

1. A inter-process communications protocol configuration system comprising:
 - a mapping of server-side protocol stack components to client-side protocol stack components in a server-side protocol stack;
 - a listing of dependent ones of said server-side protocol stack components in said server-side protocol stack based upon said mapping; and,
 - a publishable interface to said listing.
2. The system of claim 1, wherein said mapping comprises:
 - a set of linkages between protocol stack components of said server-side protocol stack and corresponding protocol stack components required to exist in an associated client-side protocol stack; and,
 - a set of attributes for selected ones of said corresponding protocol stack components.
3. The system of claim 1, wherein said listing is a reverse listing comprising a reverse ordered list of said dependent ones of said server-side protocol stack components in said server-side protocol stack based upon said mapping.
4. The system of claim 1, wherein said publishable interface is a native directory interface to at least one object incorporating said listing.

5. A method for configuring a client-side communications protocol stack, the method comprising the steps of:

consulting a mapping to determine a set of client-side protocol stack components which correspond to a set of protocol stack components in a server side protocol stack instance;

creating a listing of said determined set of client-side protocol stack components; and,

publishing said listing for access by externally disposed client computing processes.

6. The method of claim 5, wherein said consulting step further comprises the step of determining at least one attribute to be applied to at least one of said client-side protocol stack components when enabling a client-side protocol stack to interoperate with said server-side protocol stack instance.

7. The method of claim 5, wherein said creating step further comprises the step of reversing an order of said created listing.

8. A method for configuring a client-side communications protocol stack, the method comprising the steps of:

accessing a listing of required client-side protocol stack components which correspond to protocol stack components aggregated in a server-side protocol stack instance;

determining from said listing whether a client-side protocol stack instance already exists which includes said required client-side protocol stack components; and,

if said client-side protocol stack instance already exists, re-using said client-side protocol stack instance to communicate with said server-side protocol stack, and otherwise creating a new instance of a client-side protocol stack using said required client-side protocol stack components.

9. The method of claim 8, wherein said accessing step comprises the step of accessing said listing through a native directory interface.

10. The method of claim 8, further comprising the steps of:

identifying attributes to be applied to said required client-side protocol stack components; and,

applying said identified attributes to said client-side protocol stack components in said new instance.

11. The method of claim 10, wherein said re-using step comprises the steps of:

re-using said client-side protocol stack instance to communicate with said server-side protocol stack only if said client-side protocol stack instance includes said identified attributes; and,

otherwise creating a new instance of a client-side protocol stack using said required client-side protocol stack components and including said identified attributes.

12. A machine readable storage having stored thereon a computer program for configuring a client-side communications protocol stack, the computer program comprising a set of machine readable instructions which when executed by a machine method cause the machine to perform the steps of:

consulting a mapping to determine a set of client-side protocol stack components which correspond to a set of protocol stack components in a server side protocol stack instance;

creating a listing of said determined set of client-side protocol stack components; and,

publishing said listing for access by externally disposed client computing processes.

13. The machine readable storage of claim 12, wherein said consulting step further comprises the step of determining at least one attribute to be applied to at least one of said client-side protocol stack components when enabling a client-side protocol stack to interoperate with said server-side protocol stack instance.

14. The machine readable storage of claim 12, wherein said creating step further comprises the step of reversing an order of said created listing.

15. A machine readable storage having stored thereon a computer program for configuring a client-side communications protocol stack, the computer program

comprising a set of machine readable instructions which when executed by a machine method cause the machine to perform the steps of:

accessing a listing of required client-side protocol stack components which correspond to protocol stack components aggregated in a server-side protocol stack instance;

determining from said listing whether a client-side protocol stack instance already exists which includes said required client-side protocol stack components; and,

if said client-side protocol stack instance already exists, re-using said client-side protocol stack instance to communicate with said server-side protocol stack, and otherwise creating a new instance of a client-side protocol stack using said required client-side protocol stack components.

16. The machine readable storage of claim 15, wherein said accessing step comprises the step of accessing said listing through a native directory interface.

17. The machine readable storage of claim 15, further comprising the steps of:

identifying attributes to be applied to said required client-side protocol stack components; and,

applying said identified attributes to said client-side protocol stack components in said new instance.

18. The machine readable storage of claim 17, wherein said re-using step comprises the steps of:

re-using said client-side protocol stack instance to communicate with said server-side protocol stack only if said client-side protocol stack instance includes said identified attributes; and,

otherwise creating a new instance of a client-side protocol stack using said required client-side protocol stack components and including said identified attributes.